Personalized Learning in a Virtual Hands-on Lab Platform for Computer Science Education

Yuli Deng, Duo Lu, Chun-Jen Chung, Dijiang Huang and Zhen Zeng
School of Computing, Informatics, and Decision Systems
Arizona State University
Tempe, Arizona, USA
{ydeng19, duolu, cchung20, dijiang.huang, zzeng22}@asu.edu

***Abstract*—This Innovate Practice full paper presents a cloud-based personalized learning lab platform. Personalized learning is gaining popularity in online computer science education due to its characteristics of pacing the learning progress and adapting the instructional approach to each individual learner from a diverse background. Among various instructional methods in computer science education, hands-on labs have unique requirements of understanding learner's behavior and assessing learner's performance for personalization. However, it is rarely addressed in existing research. In this paper, we propose a personalized learning platform called ThoTh Lab specifically designed for computer science hands-on labs in a cloud environment. ThoTh Lab can identify the learning style from student activities and adapt learning material accordingly. With the awareness of student learning styles, instructors are able to use techniques more suitable for the specific student, and hence, improve the speed and quality of the learning process. With that in mind, ThoTh Lab also provides student performance prediction, which allows the instructors to change the learning progress and take other measurements to help the students timely. For example, instructors may provide more detailed instructions to help slow starters, while assigning more challenging labs to those quick learners in the same class. To evaluate ThoTh Lab, we conducted an experiment and collected data from an upper-division cybersecurity class for undergraduate students at Arizona State University in the US. The results show that ThoTh Lab can identify learning style with reasonable accuracy. By leveraging the personalized lab platform for a senior level cybersecurity course, our lab-use study also shows that the presented solution improves students engagement with better understanding of lab assignments, spending more effort on hands-on projects, and thus greatly enhancing learning outcomes.**

***Index Terms*—Hands-On Lab, Personalized Learning, Learning Style, Computer Science Education, Cybersecurity Education**

I. Introduction

The age of "one-size-fits-all" has passed, and personalization becomes the new norm in almost every aspect of our daily life, including learning. Personalized learning requires the instructor to pay attention to individual characteristics, such as learning style, preference, skill level, and knowledge, so as to adopt different learning materials and instructing techniques to each student. With the advent of personal computer and the Internet, many personalized e-learning systems were developed to accommodate the diverse needs of the students in various fields. Among these systems, the majority are based on traditional Learning Management Systems (LMS), whose functionalities are centered around the learning program management, learning content delivery, and student performance assessment based on written assignments as well as exams. A few systems address the needs of collaborative learning by providing online communication features like discussion board. However, most of them do not effectively support hands-on laboratories, which play an important role in project-oriented education in STEM areas, especially, in the cybersecurity education.

Starting from the very early stage of computer science education, using hands-on labs approaches has shown its significance in developing problem-solving skills by engaging the student to apply the acquired knowledge actively. This is especially true in the cybersecurity education, wherein such hands-on laboratories the students can put the learned concepts from the classroom into practice, observe the cause and consequences of system breaches, learn from challenges, and improve skills based on their encountered mistakes. In the last decade, with the rapid development of cloud virtualization, Internet, and human-computer interaction technologies, more and more hands-on labs for computer science and cybersecurity education have been designed to fit into a cloud environment and learning materials are made available online. Platforms such as Amazon AWS Education [1], MyIT Lab by Pearson [2], Microsoft Azure in Education [3] and CloudLab [4] support students to remotely access computing resources for a hands-on lab in programming, networking, and cybersecurity. However, they are merely cloud resources used for an educational purpose, which only transform physical labs into an online virtual form, and there are no dedicated personalized learning solutions that can manage and adapt learning process for students.

In this paper, we present ThoTh Lab, a personalized learning framework for cybersecurity hands-on labs in cloud environment developed in Arizona State University for upper-level computer science courses (especially, CSE 468 Computer Network Security and CSE 548 Advanced Network Security). ThoTh Lab provides a wide range of applications, virtual machines, and network devices that can fully simulate a real-world hands-on lab in cybersecurity education, where the students can set up experiments, build solutions and retrieve

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

experiment results efficiently through the Web. In addition, we built a learning behavior analyzer, a learning style classifier, and an adaptive learning content manager that utilize machine learning models and the web page interaction data as well as virtual machine logs to understand students online-learning activities, identify their learning style, and adjust learning materials accordingly. Moreover, we constructed a student learning performance assessment and prediction module for instructors to better understand students learning progress, so as to improve students' learning experience and outcomes.

To the best of authors' knowledge, ThoTh Lab is the first of its kind, which provides a complete personalized learning solution of learning style identification, learning performance assessment, and adaptive learning content delivery for virtual hands-on laboratory-based education solutions. Our experimental results show that given students behavioral data, the proposed framework can identify learners' individual characteristics and provide an accurate assessment. Moreover, our case study shows the educational benefits in terms of enhanced learning performance, a higher level of student participation and increased satisfaction with our personalized learning framework.

The remainder of this paper is organized as follows. Section II describes related works on learning style modeling and identification approaches, educational data mining, and data mining tools for e-learning environment. Section III explains the system architecture. Section IV reports our experiment setup and case study result. Finally, we conclude the paper and discussed future work directions in Section V.

II. Related Work

A. Learning Styles Identification

As defined by Honey and Mumford [5], learning styles are "a description of the attitudes and behaviors which determine an individual's preferred way of learning. Modeling and identifying learning styles is usually considered as the start point of personalized learning. In the past, several studies have proposed to model learning styles, such as the model proposed by Honey and Mumford [5], Felder and Silverman [6]. Among these models, we have adopted the Felder-Silverman Learning Style Model (FSLSM) [6] for engineering education due to its popularity and wide reception in e-learning environment research. The FSLSM classifies students according to their position in several scales that evaluate how they perceive, process and understand learning contents. The learning styles are defined in four dimensions, and each of them is represented as a pair of distinct learning styles. The first dimension considers through which sensory channel used by the learner to perceive external information most effectively – either visual or verbal. The second dimension focuses on the learner's preferred method of processing information, either active or reflective. The third dimension considers how the learner progresses toward understanding – either sequentially or globally. The fourth dimension defines what type of information that the learner preferentially perceives – either sensory or intuitive. To identify the learning styles of a student, Felder and Silverman designed a questionnaire called "Index of Learning Styles (ILS)" [6] to assess preferences of the student in all four dimensions. The questionnaire consists of 44 multiple choices questions, 11 questions for each dimension. FSLSM is one of the most frequently cited learning style models in the research area of computer-based and network-based education systems. There are also a few learning systems that are capable of adapting learning contents according to students' learning styles. But most of the approaches use the ILS as an online questionnaire to evaluate learning preferences as the first step, then present appropriate learning materials based on the answers of students. Answering the 44 questions is a time-consuming task, and consequently, it hurts user experience when applied directly in existing online personalized learning systems. Although the ILS questionnaire can tell us the general inclination of the learning styles of a student, the extent of the identified learning style cannot be estimated accurately because the student's answers are subjective.

Data-driven learning styles identification has been applied in several studies for personalizing learning. Liyanage et al. [7] presented a comparison of several data mining algorithms to detect student learning styles in a learning management system. Chang et al. [8] introduced a similar mechanism that uses k-nearest neighbor classification in an attempt to classify learning styles in a SCORM-compatible LMS environment. Kolekar et al. [9] used web usage mining and artificial neural network to identify students' learning styles in a web-based LMS environment and created an adaptive user interface for it. Garf et al. [10] [11] applied a simple rule-based student modelling approach to detect learning styles in a study with 127 students. Villaverde et al. [12] used feed-forward neural networks to detect learning styles. However, most of these researches adopt FSLSM model with traditional online learning management systems, which does not provide noticeable support of remote hands-on labs for computer science education. Learning styles will have more influence on hands-on labs, as in such situation, the students are required to be actively engaged, self-motivated, and well-paced without supervision.

B. Student Performance Prediction

Student performance prediction is another important technology that facilitates personalized instructing for teachers. A few existing systems are able to predict students performance using data mining technologies based on students activity data and existing academic record. Barber [13] successfully predicted students performance in a computer science course with data obtained from a learning management system and student profile. Myller [14] employed linear regression to predict students exam results base on source data of 103 variables collected during a classroom environment. Kotsiantis [15] expand the scope of the data source by monitoring twenty different types of log data and combining several regression techniques in order to predict students grades in a remote education program. All studies described above attempted to predict learning performance by exploiting correlations between various features and the final scores of students.

On the other hand, data mining techniques were also applied to develop performance prediction models. CalvoFlores [16] demonstrates the capability to predicted passing or failing grade for a student in an online education program using neural network models based on log data captured by a MOOC platform [17]. Researchers also have performed a wide range of comparison of various machine learning models on predicting students learning performance. From an education practitioner's perspective, neural network and SVM models are black-box models, where the internal decision-making procedure are not interpret-able and they are not easily to implemented. Thus, instructors cannot gain much useful information other than the prediction result of a score. Other researchers studied to explore domain knowledge to improve the prediction model, such as rule-based predictor, belief network, logic programming and reasoning process. These white-box methods provide explanations for all the classification results. For example, Bayesian Networks have been used to predict students learning performance using log data [18].

III. SYSTEM ARCHITECTURE

The proposed ThoTh Lab system consists of a cloud-based virtual lab platform for cybersecurity education and a few components specially designed for the personalized learning. The virtual lab platform allows instructors and students to configure and access a lab environment of virtual machines and virtual network with maximal flexibility and easiness using remote and geographically distributed cloud resources, instead of physically setting up a few computers and plug network cables into hosts, switches, and routers that are typically required by conducting a cybersecurity hands-on lab. Such labs usually require multiple machines and special network topology for generating desired types of traffic, deploying a service, attacking a server from a different machine, and etc. As the result, it is cumbersome and error-prone to configure physical devices. However, it will be extremely fast and cost-effective to set up with virtual resources in the cloud to address the issue by using a physical lab. The lab environment of each student is self-contained and can be accessed securely through an interactive web-based GUI (shown in Fig. 1). The student can sign-in to his or her virtual machines and network devices to change configurations and run any program in order to finish the tasks required by the lab instruction. Our system also keeps logging students activities over the web-based UI as well as inside the virtual machines for further analysis and personalized adaptation. The ThoTh Lab architecture contains three layers, as shown in Fig. 2.

UI Layer: This layer presents two most important parts of a hands-on lab over the browser: (a) the virtual lab environment as virtual machines as well as virtual network, and (b) the lab materials including instructions, code snippets, explanatory text, and figures. It is mainly developed in JavaScript, and has two different UI for instructors and students. The instructor can

Fig. 1. Web-based user interface of ThoTh Lab (student view).

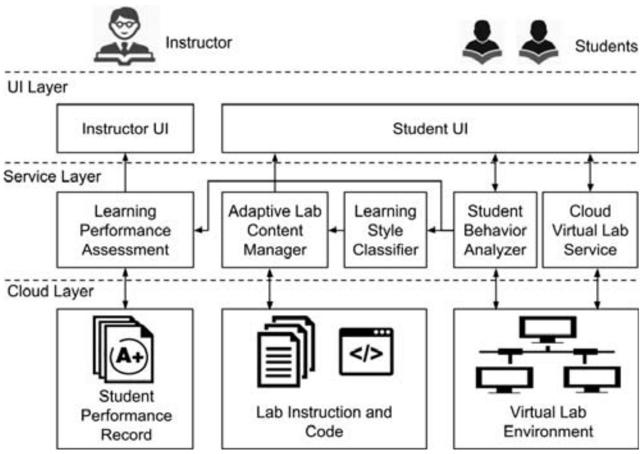

Fig. 2. System architecture of the personalized learning framework

create a hands-on lab, by setting up the lab materials using a web-based editor and configuring a virtual lab environment by dragging and dropping virtual machines and virtual network devices into the canvas. Once the lab is created, it can be submitted to the back-end cloud virtual lab management services and allow students to enroll and practice. For students, they can read the lab materials (like Fig. 3) and access the virtual machines and network devices (like Fig. 1) through the browser.

Service Layer: This layer glues the user interaction and back-end virtual machine, manages virtual resources, and provides services for certain functions for personalized learning. It is mainly implemented in PHP and Python in this project. We leveraged our previous experience on microservice architecture [19] and segment the system function into a few self-contained services. In this layer, the system (a) monitors user activities on the web UI and inside the virtual machine, (b) extracts high-level features from raw activity data, and (c) trigger learning style identification as well as lab content adaptation. More details are provided in later subsections.

Cloud Layer: This layer manages the cloud infrastructure, back-end services, user data, and lab materials. The cloud infrastructure of ThoTh Lab is built upon OpenStack [20], which is a widely used open-source cloud computing infrastructure platform. The back-end services contain various internal services for administration and management purposes. In addition, we host a repository of lab content with instructions and code in this layer using MongoDB [21], where the lab content can be flexibly adapted in different formats according to different learning styles. We provide around 60 labs created and maintained by ourselves. Some of the lab contents are rearranged and edited from SEED Labs [22], and others are written by us from scratch covering emerging topics in the cybersecurity area, including our recent research such as attacking gesture-based authentication system [23], [24], defending DDOS attack [25], deploying IDS in SDN environment [26]. Moreover, we also store students performance record (*i.e.*, lab scores graded by the instructor) in a secure database in this layer.

The remainder of this section mainly focuses on the components of the service layer in Fig. 2 that facilitate personalized learning of hands-on labs.

A. Students Behavior Analyzer

The students behavior analyzer is responsible for recording and understanding user behaviors based on low-level events such as simple features such as mouse click, mouse hover, command line activity and time spent inside a virtual machine, etc. It has three subcomponents. First, a JavaScript-based user behavior logger is implemented on the web page to monitor user's online activity. Second, a Logstash forwarder is installed inside each virtual machine of the student's lab environment to gather syslog, command line, and other activities. Third, the logged data is regularly analyzed a Python program to extract high-level students behavior features in a particular lab. Such features include session active time, lab requirement view time, and other features which has a potential correlation to the learning styles. Even though the same activity may repeat, the purpose of activity could be different depending on the lab context. Hence, it is necessary to collect and accumulate various activity patterns with the associated context for further learning performance assessment so as to examine the user's behaviors and deduce the patterns of users' meaningful behaviors.

B. Learning Style Classifier

The learning style classification module takes the output of user behavior analyzer (i.e., students behavior features), and use data mining models to identify different kind of users based on the FSLSM model discussed in section II. Before construction of the data mining models, we need to select the features that are worth modeling and useful in classification. Though data mining methods like SVM classification do not require to understand the meaning of each feature, we pick a few features based on common sense. For example, to determine whether the student prefers reflective or active learning, we analyze his/her participation in discussion systems and chatting service. For discussion forum, we analyze how often the student opens a new discussion, replies other students' message, and reads the topics posted by other students. We also collected general data which we believe has an implicit correlation with learning styles that may help our learning style classification, like mouse clicking counts, keyboard inputs and syslog events in each virtual machine. The following features are finally used.

(1) Mouse clicks count within Virtual Machine window.

(2) Keyboard inputs count within Virtual Machine.

(3) Virtual Machine syslog events' timestamps (when match with pre-defined event list).

(4) Virtual Machine bash history file (when match with pre-defined command list).

(5) Timestamps when user access, exit lab content document, play videos and click on the content navigation bar for each lab.

(6) Hint bottom access counts during lab.

(7) Quiz grades after each lab.

(8) Group Chatting message counts during lab.

(9) Discussion board topic access timestamps.

(10) Discussion board new topic publishes count and replies counts.

After feature selection, a combination of SVM [27] and Decision Tree [28] models are then used for learning style classification. We collect the data and labeled the learning style using ILS Questionnaire [29] to train the classifier, detailed in section IV. In total there are four different ways of categorizing the learning styles, and hence, ThoTh Lab has four independent sets of classifiers for these four identification tasks. It is well-known that ensemble of classifiers can improve the performance compared to using only individual constituent classifier. In particular, we combine SVM and decision tree in our framework, as there is such a big difference in the fundamental model structure between SVM and decision tree. Also, both methods have shown good compatibility and performance in related applications. In our ensemble algorithm, the first step is to construct the constituent SVM classifier and the decision tree classifier from the training dataset. Then the testing data is classified by both algorithms independently. The final predicted label is derived from the output of each constituent classifiers. If both classifiers output the same label, the label will be kept as the result. Otherwise, the framework runs the following steps:

1) If one of the prediction models classified the testing sample as neutral, neutral will be keep the label of classification.
2) Find P_{SVM}=$n(Err_{DT})$/$n(A_{SVM})$, where $n(Err_{DT})$ is the total number of training data, whose class label predicted by SVM is correct, and decision tree prediction is incorrect. $n(Asvm)$ is the total number of training data whose class label predicted by SVM is correct.
3) Find P_{DT}=$n(Err_{SVM})$/$n(A_{DT})$, where $n(Err_{SVM})$ is the total number of training data, whose class label predicted by decision tree is correct, but SVM prediction is incorrect. $n(A_{DT})$ is the total number of training data whose class label predicted by decision tree is correct.

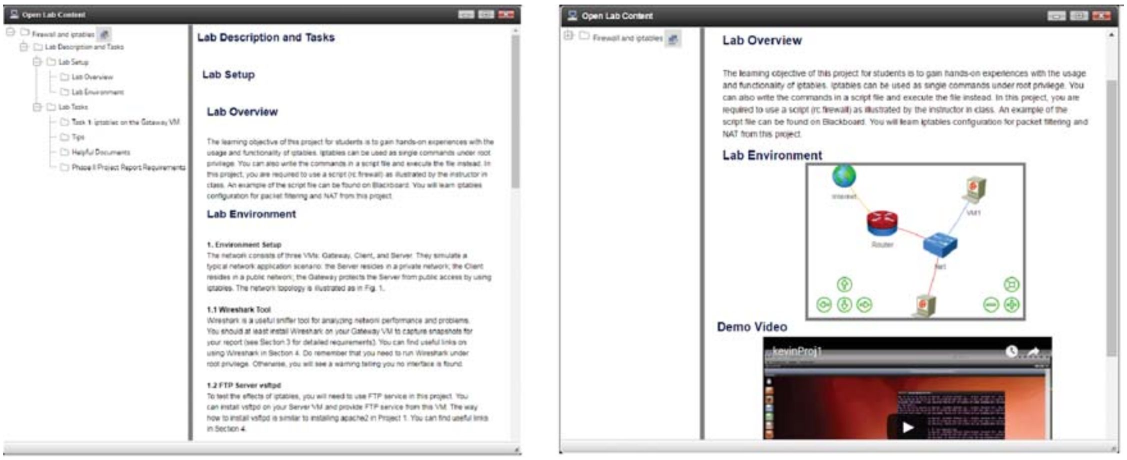

Fig. 3. Personalized lab materials for a Verbal/Sequential learner (left) and a Visual/Global learner (right).

4) Find $min(P_{SVM}, P_{DT})$, then choose class label from that classifier.

In summary, this method calculates the error rate of each classifier base on the training data and trust the classifier that makes less error will do the classification better in prediction on future data. One exception is about the neutral case, due to the fact that neutral is usually the dominant class in all four domains. Also, we observed that the accuracy of neutral label prediction by either classifier is higher than the non-neutral label prediction.

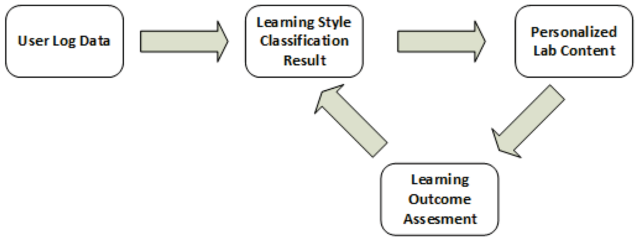

Fig. 4. Personalization Process with Feedback Path

C. Adaptive Lab Content Manager

After identifying the learning styles of students, the adaptive learning content manager will update the lab content on the web UI by selecting and constructing suitable format of lab materials from lab content repository. To change the learning content according to the identified learning style, the lab content repository should contain rich format of learning materials organized and annotated by the instructor in a way that supports such adaptation. For instance, our system will pick text-based lab instructions for a Verbal student, while adding more picture or even videos to a Visual student's lab content. Similarly, a detailed step by step lab instruction (either text or video) will be provided to a Sequential student, while a Global student will only get a one-paragraph explanation of the whole lab process. This module also obtains feedback from the student learning performance assessment module. In a situation where a student's learning performance is negatively impacted by adaptive UI, the system will reset UI to its original content and layout, and also send a request to learning style classifier to re-classify this student's learning style in next lab session again.

Fig. 3 shows two examples of how the personalized learning framework will construct lab instruction of the same lab for students with different learning styles. As the figures present, extremely detailed step-by-step instruction in a logical order was provided for a Verbal/Sequential student, while the Visual/Global student receives more abstract multimedia materials in a top-down form. One thing to note is that Active/Relative dimension was not considered in this example, as lab materials are not directly related to student's preference of processing information.

D. Learning Performance Assessment and Prediction

The personalization process in our framework uses a progress monitoring mechanism to validate whether the personalized lab environment is able to deliver effective results. If the personalized results are unfavorable, appropriate revisions must be made to personalization in order to achieve the desired learning performance. Hence, assessment feedback is crucial in this process. In order to achieve such a feedback loop as shown in Fig. 4, we developed a learning performance assessment and prediction module. This module contains three sub-components and requires a bit of assistance from the instructor. First, a JavaScript program is built to match user input command line with the requirement of a specific lab to monitoring user progress. Second, an online post-lab quiz is constructed. The quiz will ask students 10 questions randomly chosen from a question set developed by the instructor for each lab, so as to obtain some information about students' learning gain. Third, a report submission and grading assistant system are set up to collect students' lab report after each lab session and provide rough assessment and grading advise for instructors and graders. Data collection for learning performance assessment module is much more straightforward. We collected command line and syslog frequently to allow the module to estimate students progress. Then, after each lab session, we obtained quiz results and report grading estimates to determine the effectiveness of the proposed personalized system. By analyzing output from these modules, our framework constructs a feedback loop to continuously enhance and improve the performance.

The prediction part of this module takes the output of real-time assessment module and student Behavior Analyzer to estimate students' future learning performance. The Naive Bayes classification algorithm was used to predict the student performance in later semester based on earlier semester result and students behavior. A Naive Bayes classifier is a simple probabilistic classifier founded on relating Bayes theorem by naive impartiality assumptions. It is easy to build and particularly useful for medium size datasets. Three reasons we choose to use Naive Bayes model are 1. High performance when identifying at-risk students4 2. Naive Bayes model is quick to build and fast to run, and hence, it makes timely prediction possible in our system. 3. Naive Bayes algorithm is also adaptive to multiclass prediction feature, which best suits to our students log data sets. One important aspect of learning performance prediction is to identify at-risk students early. Our prediction model can be used as an early warning system to identify at-risk students in a course and inform the instructor as early as possible. Instructors will then be able to use a variety of strategies to provide at-risk students helps for improving their performance in the course.

IV. Empirical Results

A. Experiment Setup

A field experiment of our ThoTh Lab was conducted on an upper-undergraduate-level class during the 2017 Spring semester in a public university in the United States. This particular course is on network security and involves 5 hands-on labs about practical network configuration with the usage of basic network security concepts, case studies on attack and defense, and useful tools for reconnaissance and penetration. 103 senior undergraduate students registered the course, and all of them finished the ILS Questionnaire before the first lab to provide an estimation of the ground truth of their learning style preference. During the semester, each student was asked to finish first three labs in an environment on their own personal computer, then two more labs in our proposed personalized virtual hands-on lab environment. All five labs are based on the same topic and the same contents were used in the previous semester, with only minor modification to prevent cross semester plagiarism. Additional lab materials in video and picture and different formats are made to enable adaptive learning content for different learning style. For the first three labs, all students were asked to record how much time they spent on each lab. For the other two labs, students' activities were recorded online and inside the virtual machine.

TABLE I
ILS Questionnaire Results

Learning Style	# of Students	pct.	Learning Style	# of Students	pct.
Visual	30	29%	Active	20	19%
Neutral	59	57%	Neutral	61	59%
Verbal	14	14%	Reflective	22	21%
Total	103	100%	Total	103	100%
Sensory	31	30%	Global	24	23%
Neutral	40	39%	Neutral	51	50%
Intuitive	42	41%	Sequential	28	27%
Total	103	100%	Total	103	100%

B. Data Collection

During our study, various types of data were extracted from the interactions between the student and the Web-based education system. The data we were able to record and measure generally depends on the capability of the personalized virtual hands-on lab system. Thanks to the system's web and cloud nature, it's not difficult for us to capture all the web page activity of each student and Linux system log from each virtual machine they've used.

We list two uncommon features we collect during students' lab period and the motivation of choosing them. The first feature is hint link access counts. There was a hint bottom next to each section of Lab 4 content and students are allowed to click them to get help on their next lab task. The more times they click on the hint bottom, the more detail the hint will be. There are 3 levels of hints for each hint bottom. We designed this feature on purpose to identify students who have difficulty understanding and completing each lab task independently. The second uncommon feature we collect video viewing timestamps when students start and stop view guidance videos in lab content. We want to collect this in order to find those students who prefer visual learning material over traditional text-based guidance.

C. Experiment Result and Discussion

In this subsection, we'll discuss both learning style identification and learning performance prediction results. For learning style identification part, we used the distribution of students learning style identified by ILS questionnaire as the verification data for our learning style classifier (shown in Table I). We then used students' learning behavior log from the 4th lab of the semester to train and test the three classifiers in learning style classification module, as shown in Table II. We used 10-fold cross validation method to calculate the accuracy rate of the classifier output in each learning style category.

As expected, using the ensemble of classifiers results in an acceptable accuracy in all four dimensions, as it always selects the classifier with lowest misclassification rate in a particular category. Our classifier's major performance gain is on the neutral label prediction, compared with either SVM or Decision Tree method. This is directly caused by the special design of the label prediction algorithm that gives neutral label

TABLE II
LEARNING STYLE CLASSIFICATION RESULTS

Learning Style	Classification Accuracy	Learning Style	Classification Accuracy
Visual	75.0%	Active	64.5%
Neutral	83.6%	Neutral	69.7%
Verbal	77.2%	Reflective	52.4%
Total	80.6%	Total	68.0%
Sensory	68.7%	Global	66.7%
Neutral	80.4%	Neutral	91.5%
Intuitive	69.6%	Sequential	71.4%
Total	74.8%	Total	81.6%

TABLE III
LEARNING PERFORMANCE PREDICTION RESULTS

Student Category	Prediction Accuracy
Good Students(Grade A or above)	82.1%
Average Students(Grade B and B+)	69.8%
Below Average Students(Grade B-,C+,C)	81.0%
At Risk Students(Below C)	90.9%
All Students	77.7%

more weights. However, our classifier demonstrates limited performance in the Active/Reflective category, possibly due to the lack of high-quality features for that dimension, *i.e.*, the selected features are not well correlated to active and reflective learning style. On the other hands, thanks to high-quality features like video viewing time-stamps, our classifier works well in the Visual/Verbal category. The ensemble of classifiers performs worse than individual constituent classifiers in a few special cases. For example, the ensemble method returned an accuracy rate of 0.687 when identifying sensory learner, while DT and SVM returned 0.75 and 0.792 respectively. Still, the performance improvement of the ensemble of classifiers in identifying neutral learner in all dimensions is more than enough to cover the loss.

For learning performance prediction, we use the distribution of students' final lab grades and final course grades of the semester as the verification for our system. Based on the grades, performance distribution of 103 students is presented as Good (23), Average (30), Below-Average (17) and At-Risk students (10). We then used both students' grades from lab 1-3 and learning behavior log from the 4th lab to train and test the Naive Bayes classifier in learning performance prediction module. We used 10-fold cross validation method to calculate the accuracy rate of the prediction output for each category of students. Table III shows the results.

With the benefits of fast training on small data set, our performance prediction model still yields acceptable overall accuracy rate, while providing over 90% accuracy on at-risk students detection. As we discussed earlier, identifying at-risk students is the major goal for most learning performance prediction models. It is important for instructors to identify at-risk students in order to provide timely interventions. Thus,

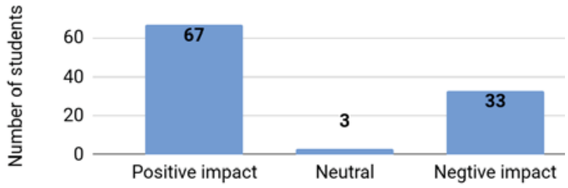

Fig. 5. Impact of personalization on individual student performance.

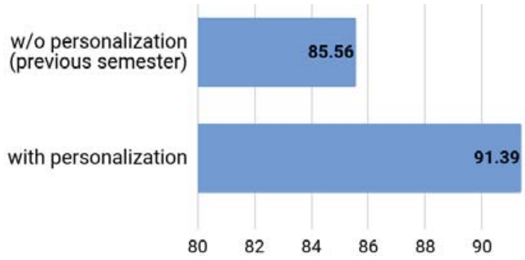

Fig. 6. Grades of same lab with and without personalization

Naive Bayes model fits our predication goal well.

D. A Case Study

We then conducted a case study using predicted learning style labels shown in Section IV.C as initial input for adaptive learning content management module for the 5th labs, and we used the feedback from learning performance assessment module and final lab grades to assess our proposed system's effectiveness on students' learning performance. Our case study result shows that majority of students achieved better grades after the utilization of personalized lab materials for their individual learning style for Lab 4 and Lab 5 (shown in Fig. 5). Among the 33 students whose performance was negatively impacted by personalized lab materials, 2 of them are students from Good category, 16 from Average category, 10 from Below-Average category and 5 from At-Risk category. Compare with the original students performance distribution, it shows that our personalized lab materials provide more positive impact on students with better performance. The average grades of Lab 5 also show improvements when be compared with the same lab from Spring 2016, which also uses the same virtual lab system, but without the personalized framework (shown in Fig. 6). Interestingly, students were inclined to spend more time on virtual labs compared to labs running on their own computers, which can be interpreted as improved engagement in the lab.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a personalized learning framework in a virtual hands-on lab platform for cybersecurity education. This framework first identifies individual student's learning styles during a lab session, then it utilizes such results to personalize lab materials for each student in future labs. Furthermore, our framework employs an automatic assessment and prediction module to closely monitor students' learning progress and performance changes in order to make adjustments to the personalize lab and notify instructor when at-risk students who need help is detected, which further improves the personalized learning framework's effectiveness. The goal of such framework is to improve students' learning efficiency as well as performance.

Our experiment and case study results show that it is possible to have a positive impact on students' learning efficiency and performance by utilizing performance prediction result and personalized content based on individual learning style. It also indicates that our framework is able to identify learning style of individual student solely based on his or her behavior in a virtual hands-on lab environment. However, our case study has limited scope. In the future, we want to conduct similar experiments to cybersecurity classes across multiple educational institutes that cover students population with much more diversity, in order to test our framework's usability and make further improvement. We also want to involve more advanced students behavior classification model such as a deep neural network with more meaningful behavior features.

ACKNOWLEDGEMENT

This research is based upon work supported by the NSF under Grants 1723440, 1642031 and 1528099, and NSFC under Grants 61628201 and 61571375.

REFERENCES

[1] "Amazon aws education," 2017. [Online]. Available: aws.amazon.com/education/awseducate/

[2] "Pearson myit lab," 2017. [Online]. Available: www.pearsonmylabandmastering.com/northamerica/myitlab/

[3] "Microsoft azure in education," 2017. [Online]. Available: azure.microsoft.com/en-us/community/education/

[4] "Cloudlab," 2017. [Online]. Available: www.cloudlab.us/

[5] P. Honey, A. Mumford *et al.*, "The manual of learning styles," 1992.

[6] R. M. Felder and L. K. Silverman, "Learning and teaching styles in engineering education," *Engineering education*, vol. 78, no. 7, pp. 674–681, 1988.

[7] M. P. P. Liyanage, K. L. Gunawardena, and M. Hirakawa, "Detecting learning styles in learning management systems using data mining," *Journal of Information Processing*, vol. 24, no. 4, pp. 740–749, 2016.

[8] Y.-C. Chang, W.-Y. Kao, C.-P. Chu, and C.-H. Chiu, "A learning style classification mechanism for e-learning," *Computers & Education*, vol. 53, no. 2, pp. 273–285, 2009.

[9] S. V. Kolekar, S. Sanjeevi, and D. Bormane, "Learning style recognition using artificial neural network for adaptive user interface in e-learning," in *Computational intelligence and computing research (ICCIC), 2010 IEEE international conference on*. IEEE, 2010, pp. 1–5.

[10] S. Graf, T.-C. Liu *et al.*, "Supporting teachers in identifying students' learning styles in learning management systems: an automatic student modelling approach," *Journal of Educational Technology & Society*, vol. 12, no. 4, p. 3, 2009.

[11] S. Graf, S. R. Viola, T. Leo, and Kinshuk, "In-depth analysis of the felder-silverman learning style dimensions," *Journal of Research on Technology in Education*, vol. 40, no. 1, pp. 79–93, 2007.

[12] J. E. Villaverde, D. Godoy, and A. Amandi, "Learning styles' recognition in e-learning environments with feed-forward neural networks," *Journal of Computer Assisted Learning*, vol. 22, no. 3, pp. 197–206, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1365-2729.2006.00169.x

[13] R. Barber and M. Sharkey, "Course correction: Using analytics to predict course success," in *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge*, ser. LAK '12. New York, NY, USA: ACM, 2012, pp. 259–262. [Online]. Available: http://doi.acm.org/10.1145/2330601.2330664

[14] N. Myller, J. Suhonen, and E. Sutinen, "Using data mining for improving web-based course design," in *International Conference on Computers in Education, 2002. Proceedings.*, Dec 2002, pp. 959–963 vol.2.

[15] S. B. Kotsiantis and P. E. Pintelas, "Predicting students marks in hellenic open university," in *Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05)*, July 2005, pp. 664–668.

[16] M.-V. A., *Current developments in technology-assisted education (2006)*. Formatex, 2006.

[17] "mooc.org." [Online]. Available: http://mooc.org/

[18] Z. A. Pardos, N. T. Heffernan, B. Anderson, and C. L. Heffernan, "The effect of model granularity on student performance prediction using bayesian networks," in *Proceedings of the 11th International Conference on User Modeling*, ser. UM '07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 435–439. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-73078-1_60

[19] D. Lu, D. Huang, A. Walenstein, and D. Medhi, "A secure microservice framework for iot," in *Service-Oriented System Engineering (SOSE), 2017 IEEE Symposium on*. IEEE, 2017, pp. 9–18.

[20] O. Sefraoui, M. Aissaoui, and M. Eleuldj, "Openstack: toward an open-source solution for cloud computing," *International Journal of Computer Applications*, vol. 55, no. 3, 2012.

[21] "Mongodb." [Online]. Available: www.mongodb.com/

[22] "Seed labs," 2002. [Online]. Available: www.cis.syr.edu/~wedu/seed/

[23] D. Lu, K. Xu, and D. Huang, "A data driven in-air-handwriting biometric authentication system," in *International Joint Conf. on Biometrics (IJCB 2017)*. IEEE, 2017.

[24] D. Lu, D. Huang, Y. Deng, and A. Alshamrani, "Multifactor user authentication with in-air-handwriting and hand geometry," in *2018 International Conference on Biometrics (ICB)*. IEEE, 2018.

[25] A. Alshamrani, A. Chowdhary, S. Pisharody, D. Lu, and D. Huang, "A defense system for defeating ddos attacks in sdn based networks," in *Proceedings of the 15th ACM International Symposium on Mobility Management and Wireless Access*. ACM, 2017, pp. 83–92.

[26] C.-J. Chung, P. Khatkar, T. Xing, J. Lee, and D. Huang, "Nice: Network intrusion detection and countermeasure selection in virtual network systems," *IEEE transactions on dependable and secure computing*, vol. 10, no. 4, pp. 198–211, 2013.

[27] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.

[28] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[29] "Ils questionnaire." [Online]. Available: www.webtools.ncsu.edu/learningstyles/